Advanced Regression: 3b Penalised regression models (ridge, lasso, elastic net)

Garyfallos Konstantinoudis

Epidemiology and Biostatistics, Imperial College London

7th March 2023

The linear model

$$y = \alpha + x\beta + \epsilon$$

- y: Outcome, response, dependent variable
  Dimension: $n \times 1$
- x: Regressors, exposures, covariates, input, explanatory, or independent variables
  Dimension: $n \times p$
- ϵ: Residuals, error
- α: Intercept
- β: Regression coefficients, vector of length p

Classical regression

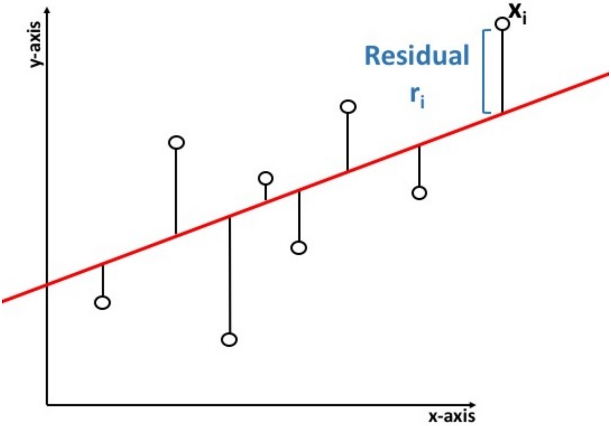
- The ordinary least squares $\hat{\beta}_{OLS}$ is defined as

$$\hat{\beta}_{OLS} = \underbrace{(x^t x)^{(-1)}}_{p\times p} \underbrace{x^t}_{p\times n} \underbrace{y}_{n\times 1}.$$

- The residual sum of squares (RSS) is minimised by the ordinary least squares estimate

$$\begin{aligned} RSS(\alpha, \beta) &= \epsilon_1^2 + \ldots + \epsilon_i^2 + \ldots + \epsilon_n^2 \\ &= \sum_{i=1}^{n} \epsilon_i^2 \\ &= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \\ &= \sum_{i=1}^{n} (y_i - (\alpha + \beta x_i))^2 \end{aligned}$$

Residual sum of squares (RSS)



- Note $\sum_{i=1}^{n} \epsilon_i = 0$

Classical regression and high-dimensional data

- When $n << p$ the ordinary least squares cannot be computed because $\underbrace{(x^t x)}_{p \times p}$ is singular (rank n).
- Bias-variance trade-off:
  - The ordinary least squares estimate is best linear unbiased estimator (BLUE).
  - BEST (smallest variance) among UNBIASED (zero bias) estimators.
  - When considering high-dimensional data, the ordinary least squares estimate has a high variability. (dramatically different over different samples).
  - We rather prefer an estimate that is biased (towards a sensible option, e.g. the Null), but is precise, (ie has low variance).

IDEA: Control the estimates' variance by not allowing the to be too big. Constraints on how big they get. Does it remind you anything?

Motivation for penalised least squares

Minimise RSS but with penalty

$$\underset{\alpha,\beta}{argmin} = \underbrace{RSS(\alpha, \beta)}_{\text{Residual Sum of Squares}} + \underbrace{\lambda f(\beta)}_{\text{penalty}}$$

where

- Residual Sum of Squares: $RSS(\alpha, \beta) = \sum_{i=1}^{n} (y_i - (\alpha + \beta x_i))^2$
- Penalty term as a function of the regression coefficients β: $f(\beta)$
- Regularization parameter: λ
- The intercept is not penalised

Motivation for penalised least squares

The penalty introduces a bias, so why do it?

- Which variables do we include? Only those for which it is worth to take the penalty.
- Occam's razor: It induces sparsity and favours models with lower complexity (Lasso and elastic net).
- Regularizes the inversion of $x^t x$ (Ridge regression).

Different penalty terms define different methods

$$\underset{\alpha,\beta}{argmin} = RSS(\alpha, \beta) + \lambda f(\beta)$$

- Ridge regression: L2 penalty

$$\lambda f(\beta) = \lambda \sum_{j=1}^{p} \beta_j^2$$

- Lasso regression: L1 penalty

$$\lambda f(\beta) = \lambda \sum_{j=1}^{p} |\beta_j|$$

- Elastic net regression: L1 + L2 penalty

$$\lambda f(\beta) = \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2$$

Ridge regression

Ridge regression uses the L2 norm as penalty:

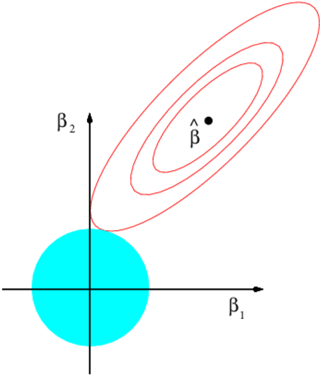
$$\underset{\alpha,\hat{\beta}_{Ridge}}{argmin} = \underbrace{RSS(\alpha,\beta)}_{\text{Residual Sum of Squares}} + \underbrace{\lambda\sum_{j=1}^{p}\beta_j^2}_{\text{penalty}}$$

Interpretation:

- The Ridge regression coefficient $\hat{\beta}_{Ridge}$ is a biased estimate, but has a reduced variance compared to $\hat{\beta}_{OLS}$.
- There is no intrinsic model selection in Ridge regression, all p variables will have $\hat{\beta}_{Ridge} \neq 0$.
- Minimise the RSS while forcing β not to be very large.

Ridge regression: Geometric interpretation

- $||\beta||_2^2 \leq c^2$
- Where is the BLUE?
- Where is the ridge solution?



Ridge regression

$$\sum (Y_i - \alpha - \beta_1 x_i - \ldots)^2 \text{ subject to } ||\beta||_2^2 \leq c^2$$

$$F(\alpha, \beta, \lambda) = \sum (Y_i - \alpha - \beta_1 x_i - \ldots)^2 + \lambda(\beta_1^2 + \beta_2^2 + \cdots - c^2)$$

How can we solve it?

- Partial derivatives
- Numerical solution using different values for λ. Note $\lambda \geq 0$

$$argmin\{F(\alpha, \beta, \lambda)\} = RSS(\alpha, \beta) + \lambda \sum_{j=1}^{p} \beta_j^2$$

Interpretation of λ

- $\lambda = 0$, then OLS
- $\lambda >> 0$, then β=0

Ridge regression and ordinary least squares

The ridge regression estimate is available in closed form

$$\hat{\beta}_{Ridge} = \underbrace{(x^t x + \lambda I)^{(-1)}}_{p \times p} \underbrace{x^t}_{p \times n} \underbrace{y}_{n \times 1},$$

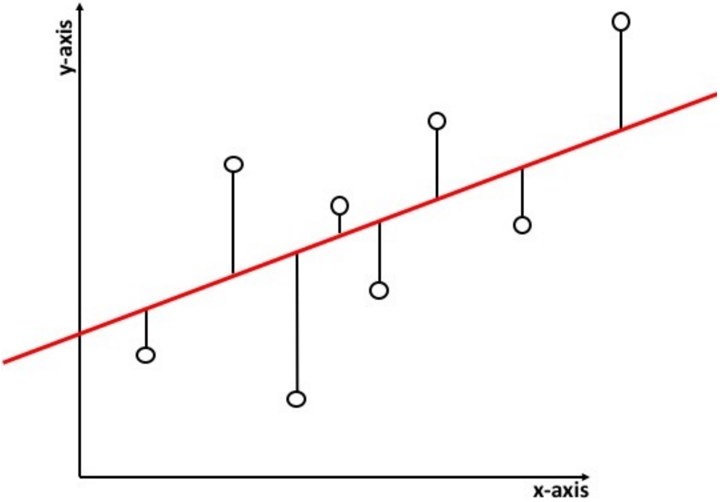
where I is a $p \times p$ diagonal matrix with ones on the diagonal and zero on the off-diagonal

$$x^t x + \lambda I = n \begin{bmatrix} cov(x_1) & cov(x_{12}) & cov(x_{13}) \\ cov(x_{21}) & cov(x_2) & cov(x_{23}) \\ cov(x_{31}) & cov(x_{23}) & cov(x_3) \end{bmatrix} + \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}.$$

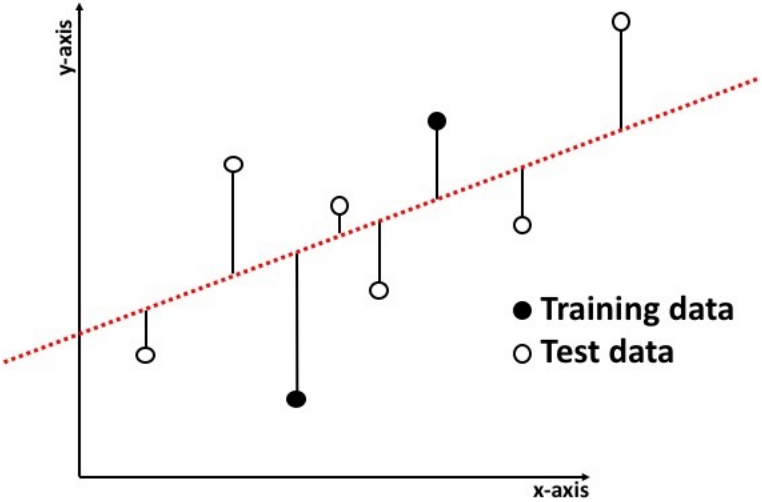
This resembles the OLS estimate apart from $+\lambda I$

$$\hat{\beta}_{OLS} = \underbrace{(x^t x)^{(-1)}}_{p \times p} \underbrace{x^t}_{p \times n} \underbrace{y}_{n \times 1}.$$

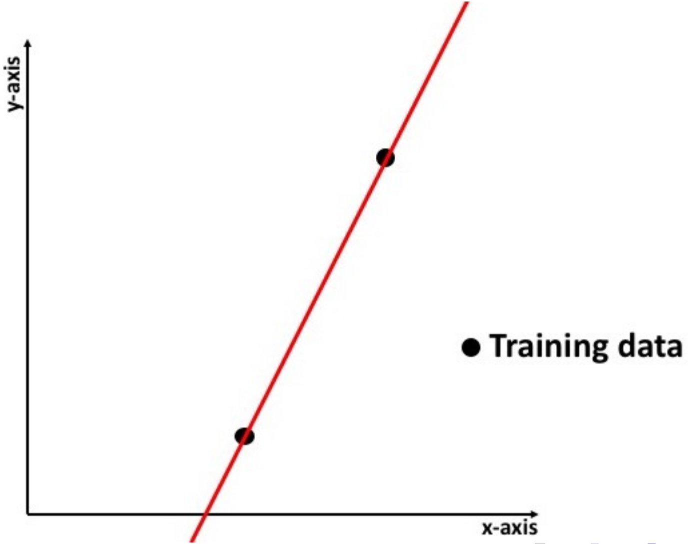
Ridge regression and ordinary least squares



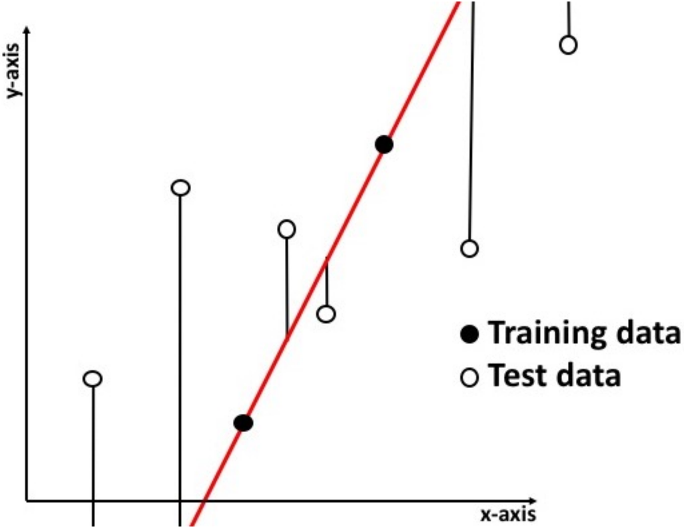
Ridge regression and ordinary least squares



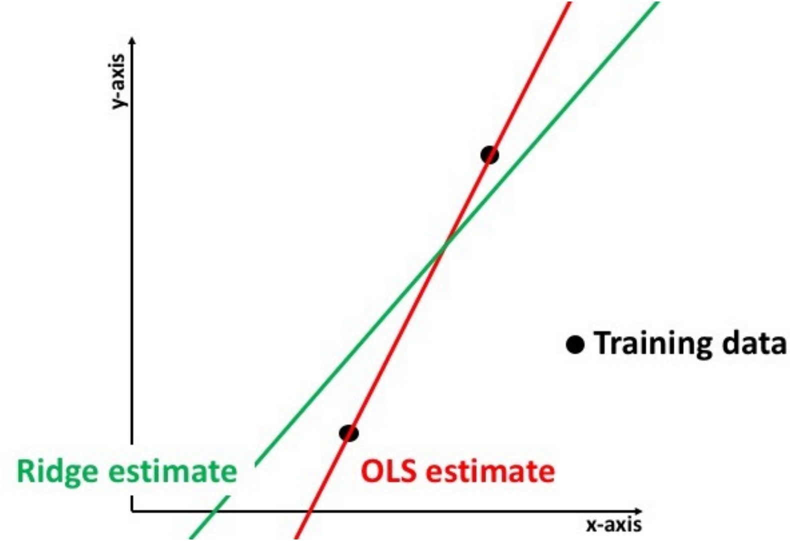
Ridge regression and ordinary least squares



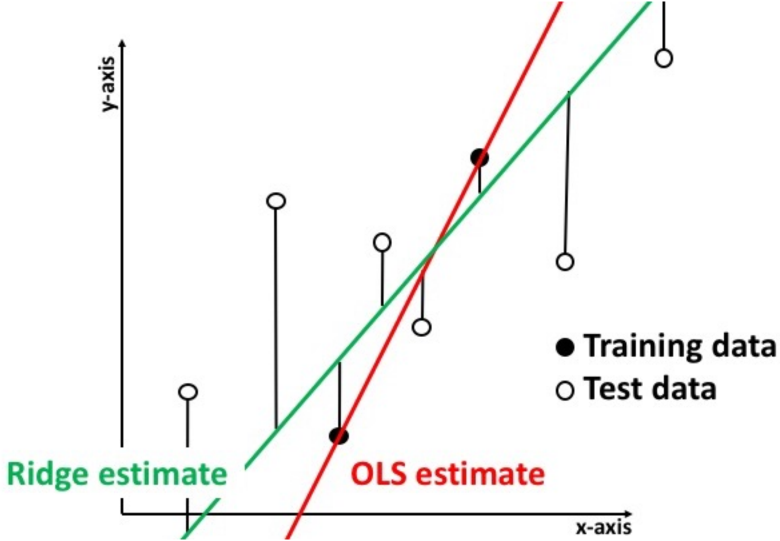
Ridge regression and ordinary least squares



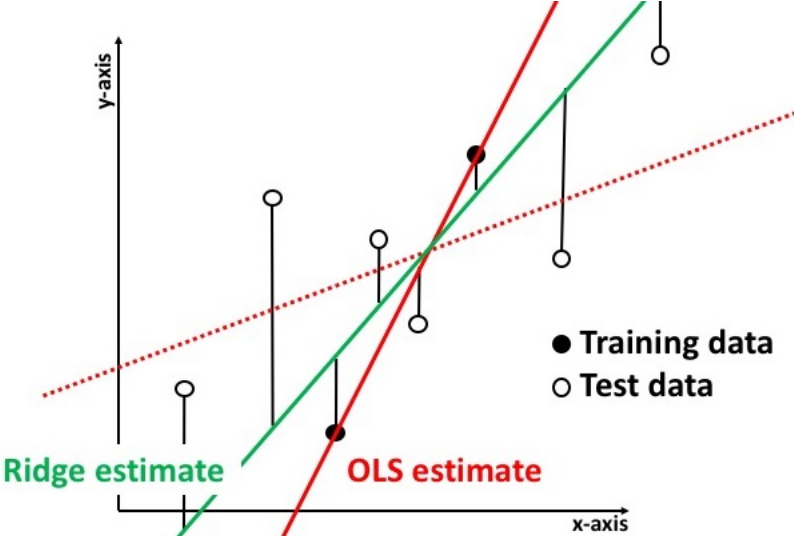
Ridge regression and ordinary least squares



Ridge regression and ordinary least squares



Ridge regression and ordinary least squares



Lasso regression

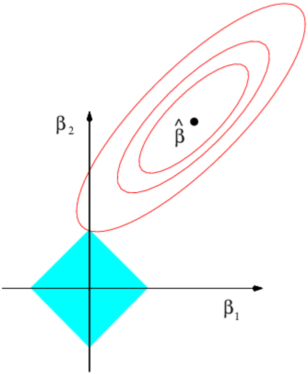
$$\underset{\hat{\alpha},\hat{\beta}_{Lasso}}{argmin} = \underbrace{RSS(\alpha, \beta)}_{\text{Residual Sum of Squares}} + \underbrace{\lambda \sum_{j=1}^{p} |\beta_j|}_{\text{penalty}}$$

Interpretation:

- The Lasso regression coefficient $\hat{\beta}_{Lasso}$ is a biased estimate, but has a reduced variance compared to $\hat{\beta}_{OLS}$.
- There is an intrinsic model selection in Lasso regression, as it sets certain variables exactly to $\hat{\beta}_{Lasso} = 0$, and thus excludes them from the model.
- When two variables are highly correlated, Lasso includes only one (at random) and not both.

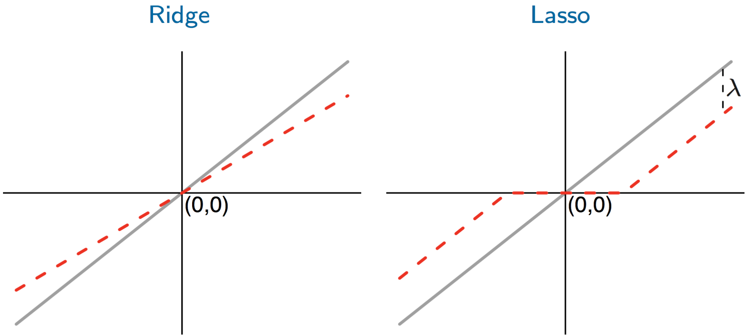
Lasso regression: Geometric interpretation

- $||\beta||_1 \leq c$
- Where is the BLUE?
- Where is the lasso solution?

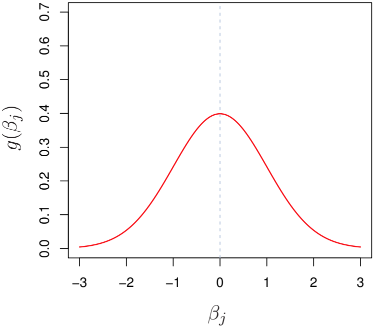


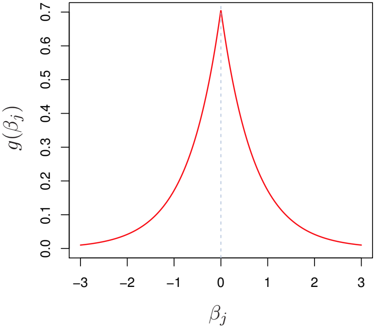
If circle, then we are not really sure where min β is, but if diamond the sharp edges makes it likely to hit at 0: $\beta_2 = c$ and $\beta_1 = 0$.

Ridge and lasso: Induced shrinkage



Ridge and lasso: Bayesian interpretation





- Left: Ridge regression is the posterior mode for β under a Gaussian prior.
- Right: Lasso regression is the posterior mode for β under a double-exponential prior.

Elastic net regression

$$\underset{\hat{\alpha},\hat{\beta}_{\text{Elastic net}}}{argmin} = \underbrace{RSS(\alpha,\beta)}_{\text{Residual Sum of Squares}} \underbrace{+\lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2}_{\text{penalty}}$$

Interpretation:

- The Elastic net regression coefficient $\hat{\beta}_{\text{Elastic net}}$ is a biased estimate, but has a reduced variance compared to $\hat{\beta}_{OLS}$.
- There is an intrinsic model selection in Lasso regression, as it sets certain variables exactly to $\hat{\beta}_{\text{Elastic net}} = 0$, and thus excludes them from the model.
- When two variables are highly correlated, Elastic net includes both (Grouping property).

Elastic net regression: reparametrization

$$\underset{\hat{\alpha},\hat{\beta}_{\text{Elastic net}}}{\mathit{argmin}} = \underbrace{RSS(\alpha, \beta)}_{\text{Residual Sum of Squares}} + \lambda \left[\alpha \, ||\beta||_1 + (1-\alpha) \, ||\beta||_2^2 \right]$$

- α can be seen as a mixing parameter
- When $\alpha = 0$ ridge regression
- When $\alpha = 1$ lasso regression
- How can we select the optimal λ and α?

How to tune the regularisation parameter?

$$\underset{\alpha,\beta}{argmin} = RSS(\alpha, \beta) + \lambda f(\beta)$$

λ is the regularisation parameter

- $\lambda = 0$: No regularisation
- Small λ: Minimal regularisation
- Large λ: Strong regularisation
- How to choose the optimal λ?
  $\rightarrow$ Cross-validation (Lecture 3c)

Prediction using penalised regression

- Regularized regression is an ideal tool for prediction.
- We can define a prediction rule $\hat{f}(x)$ using the penalised regression coefficients

$$\hat{y} = \hat{f}(x) = \alpha + x\hat{\beta}_{\text{Penalised}}$$

  where $\hat{\beta}_{\text{Penalised}}$ are the p regularized regression coefficients.
- Since Lasso and Elastic net force some $\hat{\beta}_{\text{Penalised}}$ to zero, variables with $\hat{\beta}_{\text{Penalised}} = 0$ are excluded from the model and do not contribute to the prediction rule.
- In contrast in Ridge regression all variables contribute to $\hat{f}(x)$.

Penalised regression in R: glmnet()

```
glmnet(x, y, family, alpha,nlambda = 100,
lambda.min.ratio = ifelse(nobs<nvars,0.01,0.0001),
lambda=NULL, standardize = TRUE, intercept=TRUE)
```

Input

- y: Outcome or response
- x: Predictors, formatted `as.matrix(x)`

Generalised linear models included

- Linear regression: family = 'gaussian'
- Logistic regression: family = 'binomial'
- Count regression: family = 'poisson'
- Categorical outcome: family = 'multinomial'
- Survival model: family = 'cox'
- Multivariate linear model: family = 'mgaussian'

Penalised regression in R: glmnet()

```
glmnet(x, y, family, alpha, nlambda = 100,
lambda.min.ratio = ifelse(nobs<nvars,0.01,0.0001),
lambda=NULL, standardize = TRUE, intercept=TRUE)
```

Penalised regression models:

- Ridge regression: alpha $=0$
- Lasso regression: alpha $=1$
- Elastic net: $0<$alpha<1

Regularisation parameter:

- Specify lambda for a pre-defined regularisation parameter
- Recommended: Perform cross-validation
  - nlambda: Grid length
  - lambda.min.ratio $=$ ifelse(nobs$<$nvars,0.01,0.0001)

Penalised regression in R: glmnet

glmnet.out = `glmnet(x, y, family, alpha)`

Values:

- Intercept: glmnet.out$a0
- Regression coefficient estimates: glmnet.out$beta
- Regularisation parameters used: glmnet.out$lambda

Functions:

- Cross-validation: `cv.glmnet()`
- Regression coefficients: `coef(glmnet.out)`
- Prediction: `predict(glmnet.out, newx)`

Penalised regression in R: glmnet

For more details on glmnet, see the useful vignette: `http://web.stanford.edu/~hastie/glmnet/glmnet_alpha.html`

Other packages in R

- lm.ridge in the MASS package
- lars in the lars package
- penalized in the penalized package

Example: Diabetes data

- y: quantitative measure of disease progression one year after baseline (vector)
- x: predictor matrix
  - clinical parameters: age, sex, bmi
  - map: blood pressure
  - tc: total cholesterol
  - ldl: low-density lipoprotein
  - hdl: high-density lipoprotein
  - tch: total cholesterol over hdl
  - ltg: triglycerides
  - glu: glucose
- $n = 442$: sample size

Ridge regression and diabetes data

1. `lm(y~x)`
2. `glmnet(x,y,family="gaussian",alpha=0,lambda=0.1)`
3. `glmnet(x,y,family="gaussian",alpha=0,lambda=1)`

	lm	ridge01	ridge1
(Intercept)	152.13348	152.133484	152.133484
age	-10.01220	-9.358965	-6.698824
sex	-239.81909	-238.769510	-233.325125
bmi	519.83979	520.713588	520.019917
map	324.39043	323.548886	319.639206
tc	-792.18416	-666.737488	-320.594626
ldl	476.74584	377.500940	103.343333
hdl	101.04457	45.579036	-104.230542
tch	177.06418	161.855769	124.122091
ltg	751.27932	703.885551	568.507179
glu	67.62539	68.277472	71.865726

Lasso regression and diabetes data

1. `lm(y~x)`
2. `glmnet(x,y,family="gaussian",alpha=1,lambda=0.1)`
3. `glmnet(x,y,family="gaussian",alpha=1,lambda=40)`

	lm	lasso01	lasso40
(Intercept)	152.13348	152.133484	152.13348
age	-10.01220	-5.789635	.
sex	-239.81909	-234.457334	.
bmi	519.83979	522.819506	93.58588
map	324.39043	320.347881	.
tc	-792.18416	-534.397332	.
ldl	476.74584	271.305848	.
hdl	101.04457	-9.067565	.
tch	177.06418	146.255119	.
ltg	751.27932	655.715819	33.43273
glu	67.62539	66.410644	.

Elastic regression and diabetes data

1. `lm(y~x)`
2. `glmnet(x,y,family="gaussian",alpha=0.5,lambda=0.1)`
3. `glmnet(x,y,family="gaussian",alpha=0.5,lambda=40)`

	lm	enet01	enet40
(Intercept)	152.13348	152.133484	152.13348
age	-10.01220	-7.373073	.
sex	-239.81909	-236.908421	.
bmi	519.83979	521.524719	308.42812
map	324.39043	321.784878	53.18902
tc	-792.18416	-570.166854	.
ldl	476.74584	302.943444	.
hdl	101.04457	.	.
tch	177.06418	144.752485	.
ltg	751.27932	669.554762	267.61977
glu	67.62539	67.483126	.

Example: Breast cancer data

- y: benign or aggressive tumour (binary)

Benign	Aggressive	Total
185	121	306

- x: gene expression of $p = 22,283$ genes
- $n = 306$: sample size
- Truly big data $n << p$
- Data taken from Hatzis et al 2011 `https://jamanetwork.com/journals/jama/fullarticle/899864`

Breast cancer data and glm

1. `glm(severity∼as.matrix(x), family='binomial')`

```
> glm.out = glm(severity~as.matrix(x), family="binomial")
glm.out$converged
```

```
Error: vector memory exhausted (limit reached?)
In addition: Warning message:
glm.fit: algorithm did not converge
```

```
> glm.out$converged
[1] FALSE
```

Breast cancer data and lasso

```
> lasso.out00015 = glmnet(as.matrix(x),y=severity,family="binomial",alpha=1,lambda=0.001
> sum(abs(lasso.out00015$beta)>0)
[1] 241
>
> lasso.out012 = glmnet(as.matrix(x),y=severity,family="binomial",alpha=1,lambda=0.12)
> sum(abs(lasso.out012$beta)>0)
[1] 3
>
> lasso.out012$a0
       s0
-0.229678
> summary(lasso.out012$beta)
22283 x 1 sparse Matrix of class "dgCMatrix", with 3 entries
      i j            x
1   411 1 -0.024275637
2  5307 1 -0.001092753
3 18933 1  0.008228413
```

Breast cancer data and elastic net

```
> enet.out003 = glmnet(as.matrix(x),y=severity,family="binomial",alpha=0.5,lambda=0.003)
> sum(abs(enet.out003$beta)>0)
[1] 441
>
> enet.out024 = glmnet(as.matrix(x),y=severity,family="binomial",alpha=0.5,lambda=0.24)
> sum(abs(enet.out024$beta)>0)
[1] 4
>
> enet.out024$a0
        s0
-0.2758569
> summary(enet.out024$beta)
22283 x 1 sparse Matrix of class "dgCMatrix", with 4 entries
      i j            x
1   411 1 -0.015083256
2   904 1 -0.002995031
3  5307 1 -0.001209533
4 18933 1  0.006024934
```

Take away: Penalized regression models

- Regularized regression approaches minimise the residual sum of squares and an additional penalty function.
- Different penalties imply different approaches:
  - Ridge regression: $L2$
  - Lasso regression: $L1$
  - Elastic net regression: $L1 + L2$
- Penalized regression approaches are biased, they underestimate the size of the true effect.
- But they reduce the variance of the estimate and the prediction rule.
- Lasso and Elastic net perform an intrinsic model selection.
- The regularisation parameter λ can be chosen using cross-validation.

Further reading:

- An Introduction to Statistical Learning: Chapter 6.2 (Shrinkage Methods) and 6.6 (Lab 2: Ridge Regression and the Lasso) `http://www-bcf.usc.edu/~gareth/ISL/index.html`
- The epigenetic clock: 'A multi-tissue full lifespan epigenetic clock for mice' `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6224226/` *Using DNA methylation data from previous publications with data collected in house for a total 1189 samples spanning 193,651 CpG sites, we developed 4 novel epigenetic clocks by choosing different regression models (elastic net- versus ridge-regression) and by considering different sets of CpGs (all CpGs vs highly conserved CpGs).*